Data-center networking

Malathi Veeraraghavan
University of Virginia
mvee@virginia.edu

Tutorial at IEEE ICC 2014
June 14, 2014

Web site: http://www.ece.virginia.edu/mv

Thanks to the Jogesh Muppala, HKUST.
Also thanks to US DOE ASCR for grant DE-SC0007341 &
NSF for grants, OCI-1127340, ACI 1340910, CNS-1405171, and CNS-1116081.

Outline

- Introduction
- Challenges in data center networking
- Research papers:
  - Ethernet based
  - New protocols: DCell, B-Cube
  - Optical, wireless, and energy-efficient architectures
- Standards:
  - IEEE TRILL
  - IEEE 802.1Q
- Summary

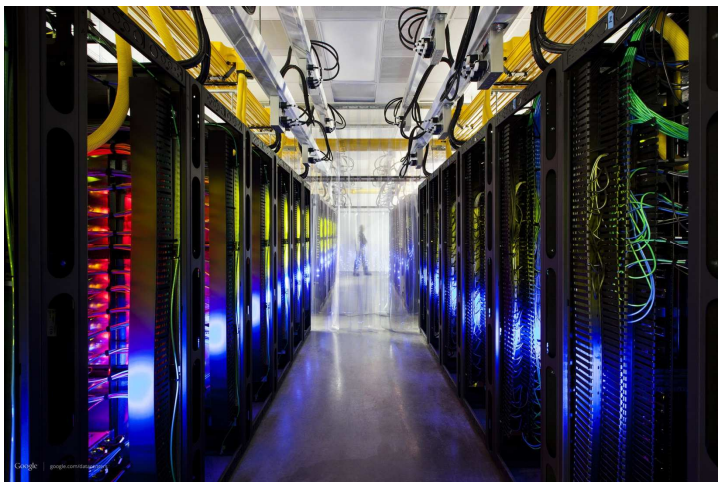
Two use cases for data centers

- Commercial data centers
  - Amazon, Google, Microsoft, Yahoo, IBM
  - Cloud applications
- Scientific data centers
  - US DOE: OLCF, ALCF, NERSC
  - US NSF: XSEDE project, NWSC
  - Scientific applications

DOE: Department of Energy; OLCF: Oak Ridge Leadership Computing Facility
ALCF: Argonne Leadership Computing Facility
NERSC: National Energy Reserch Supercomputing Center; NSF: National Science Foundation
XSEDE: Extreme Science & Engineering Discovery Environment
NWSC: NCAR Wyoming Super Computing Center; National Center for Atmospheric Research

Inside Google's Data Center

A Campus Network Room in Council Bluffs, IA Data Center

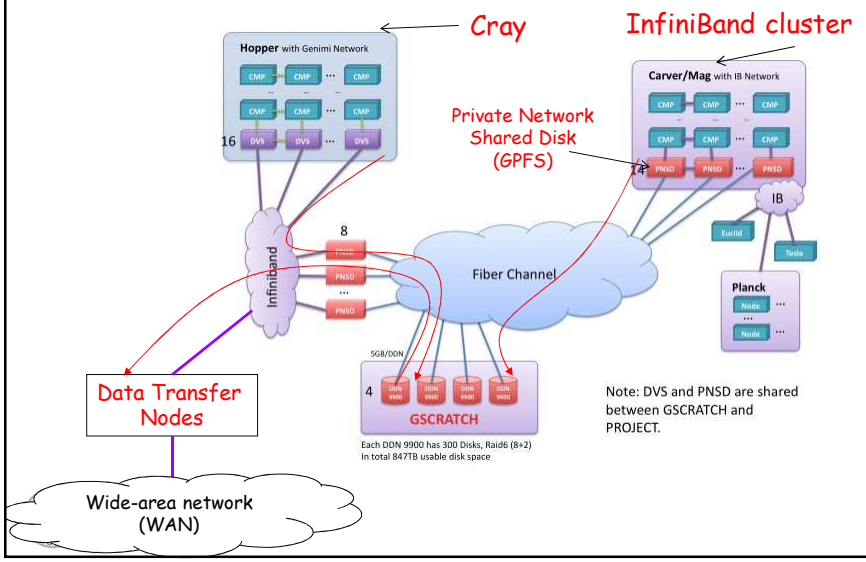


NERSC computational systems

- http://www.nersc.gov/systems/computational-systems-table/
- Edison: Cray XC300; Aries interconnect
- Hopper: Cray XE6; Gemini interconnect
- Carver: IBM iDataPlex; InfiniBand
- Genpool: DOE Joint Genome Institute
- General Parallel File System (GPFS) servers RAID arrays, and FibreChannel: Storage
- Data transfer nodes: WAN access

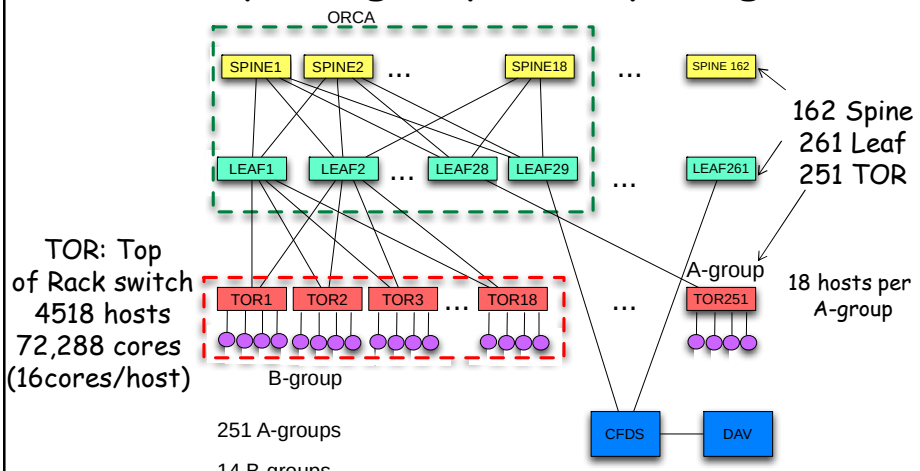
5

Example: NERSC systems



6

Yellowstone: InfiniBand Cluster
NCAR Wyoming Supercomputing Center



251 A-groups

14 B-groups

9 ORCAs

Applications

- Commercial:
  - Hadoop MapReduce, Microsoft Dryad
- Scientific:
  - Message Passing Interface (MPI)
  - Community Earth System Model (CESM)
    - POP (Ocean model), CAM (atmosphere), CICE (sea-ice), CLM (Community Land Model), and CPL (central coupler)
    - http://www.cesm.ucar.edu/models/cesm1.0/

Data Center Network Requirements

- Requirements for data center networks
  - R1: Any VM may be migrated to any physical machine without a change in its IP address
  - R2: An administrator should not need to configure any switch before deployment
  - R3: Any end host should efficiently communicate with any other end host through any available path
  - R4: No forwarding loops
  - R5: Failure detection should be rapid and efficient

Virtualization

- Virtual machines (VM): VMware, Xen
  - Allows each user to run their own OS for their apps
- Why VMs?
  - To improve server utilization
- Why is application based sharing insufficient?
  - Resource contention: need to separate out resource allocations for each user's set of applications
  - Security considerations
  - Sensitivity to OS patches and versions. If the OS is upgraded or patched to allow one app to run, an older app may stop working
- Migrate VMs for maintenance & energy savings

Outline

- Introduction
- ➢ Challenges in data center networking
- Research papers:
  - Ethernet based
  - New protocols: DCell, B-Cube
  - Optical, wireless, and energy-efficient architectures
- Standards:
  - IEEE TRILL
  - IEEE 802.1Q
- Summary

Challenges

- Neither Ethernet switched networks nor IP-routed networks are completely satisfactory for datacenter networks
- Energy consumption: 2006: for US DCs alone: consumption > 61 billion kWh
- Failure recovery: with 100K hosts, per-day failures are inevitable
- Wiring

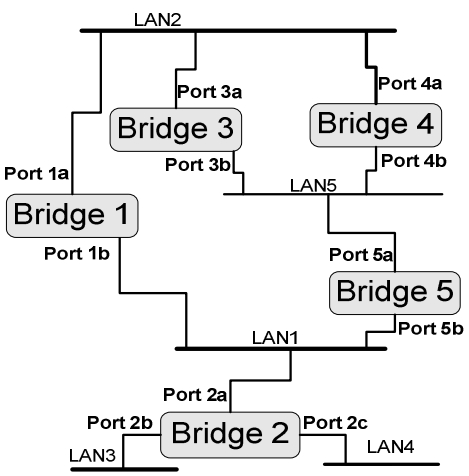
Ethernet-switched vs. IP-routed networks

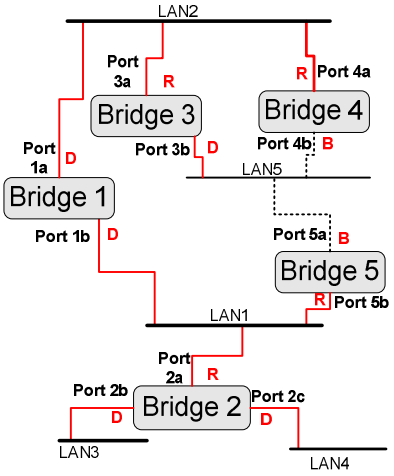
Action	Ethernet-switched networks	IP-routed networks
Address assignment required? (configuration overhead)	No; Flat addressing; a network interface card can be connected to any switch ✓	Yes; Hierarchical addressing; Topological location based address assignment; all interfaces in a subnet need to be assigned addresses with the same subnet ID ✗
VM Migration	✓	Address reconfig. required ✗
Scalability	Not good; flooding until addresses are learned ✗	Good; because of hierarchical addressing ✓
Route selection (efficient use of network links?)	Spanning tree protocol blocks ports to prevent loops ✗	OSPF, IS-IS protocols determined routing tables Equal-cost multi-path? ✓

Spanning tree protocol (STP)

- Goal: Break routing loops
- Configuration Bridge Protocol Data Units (BPDUs) are exchanged between switches
- Plug-and-play: Pre-assigned priority ID and MAC address of port 1 determine default bridge ID
- Root bridge of tree: one with smallest bridge ID
- Each bridge starts out thinking it is the root bridge
- Through BPDU exchanges, tree converges, which means all switches have same view of the spanning tree
- Each bridge determines which of its ports should be root ports and which designated ports
- These ports are placed in forwarding state; rest are blocked
- Packets will not be received or forwarded on blocked ports
- Advantage: zero-configuration!
- Disadvantages:
  - root bridge could become bottleneck
  - no load balancing

Example of STP





STP: Advantages/disadvantages

- Advantage:
  - Plug and Play – No configuration required
- Disadvantages:
  - Scalability issue:
    - Flooding used until MAC addresses learned
  - No easy loop detection methods:
    - No hop count or time-to-live in Ethernet header to drop looping packets
  - Layer 2 redundancy unexploited:
    - Blocked links created by STP

Outline

- Introduction
- Challenges in data center networking
- ➢ Research papers:
  - Ethernet based
  - New protocols: DCell, B-Cube
  - Optical, wireless, and energy-efficient architectures
- Standards:
  - IEEE TRILL
  - IEEE 802.1Q
- Summary

17

Research proposals
(new switches; Ethernet NICs in hosts)

- Kim, C., Caesar, M., Rexford, J.: Floodless in SEATTLE: a scalable Ethernet architecture for large enterprises. In: ACM Sigcomm 2008
- Al-Fares, M., Loukissas, A., Vahdat, A.: A scalable, commodity data center network architecture. ACM Sigcomm 2008 (670 citations)
- Niranjan Mysore, R., Pamboris, A., Farrington, N., Huang, N., Miri, P., Radhakrishnan, S., Subramanya, V., Vahdat, A.: Portland: a scalable fault-tolerant layer 2 data center network fabric. ACM Sigcomm 2009
- Greenberg, A., Hamilton, J.R., Jain, N., Kandula, S., Kim, C., Lahiri, P., Maltz, D.A., Patel, P., Sengupta, S.: VL2: a scalable and flexible data center network. ACM Sigcomm 2009

NIC: Network Interface Card

18

SEATTLE (arbitrary topology)

- Link-state protocol:
  - only for switch-level topology
- Store location s_a of host interface MAC_a at switch r_a determined by using hash operation $F(MAC_a)=r_a$
- When a frame destined to MAC_a arrives at switch s_b:
  - it executes the hash function $F(MAC_a)$ and finds r_a
  - then it tunnels the frame to r_a, which tunnels the frame to s_a where MAC_a is located
  - r_a notifies s_b so that future packets can be sent directly
- Consistent hashing – to avoid churn in mappings if a switch drops out of the list

Basic concept (SEATTLE)

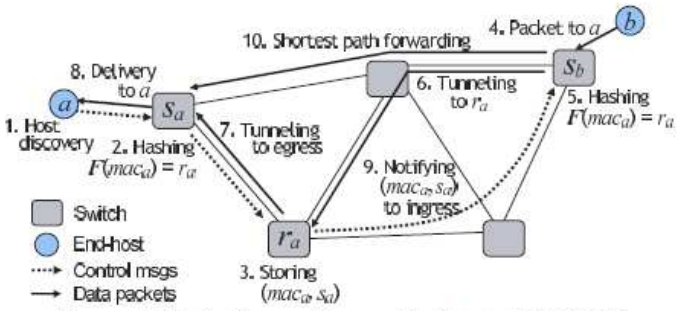
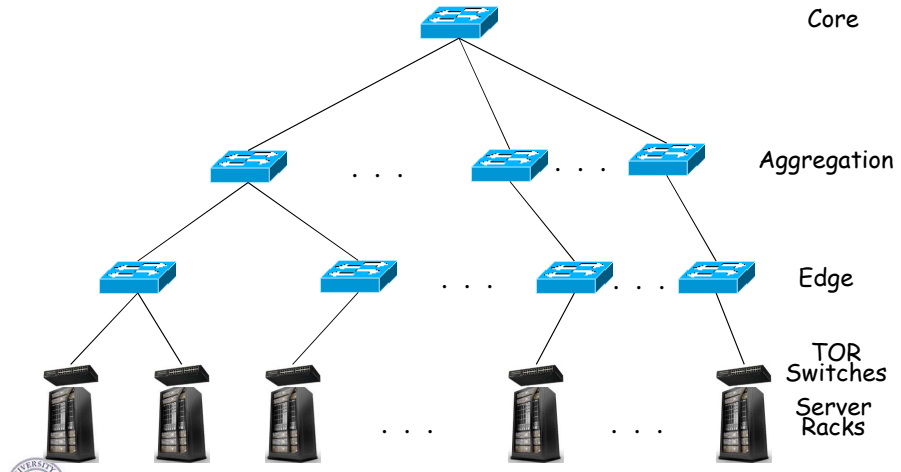


Figure 3: Packet forwarding and lookup in SEATTLE.

Drawback: Interesting research proposal but it requires brand new switch implementation

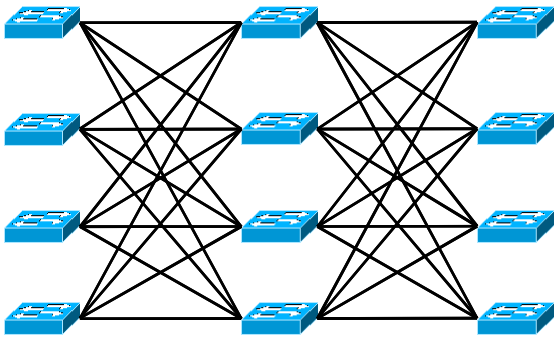
Kim, C., Caesar, M., Rexford, J.: Floodless in SEATTLE: a scalable ethernet architecture for large enterprises. In: ACM SIGCOMM Computer Communication Review, vol. 38, pp. 3–14. ACM (2008)

Basic Tree Topology



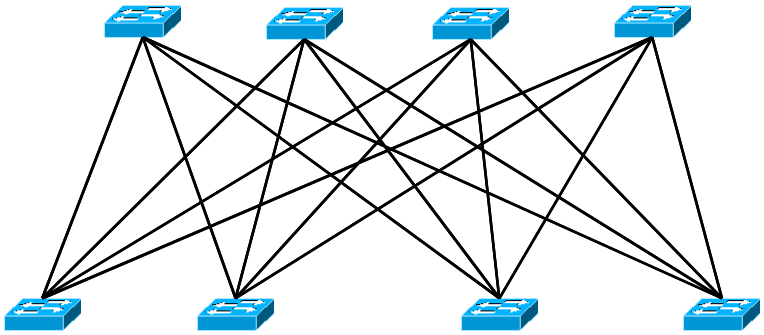
Clos Networks

- 3-stage Clos (used for switch fabrics)



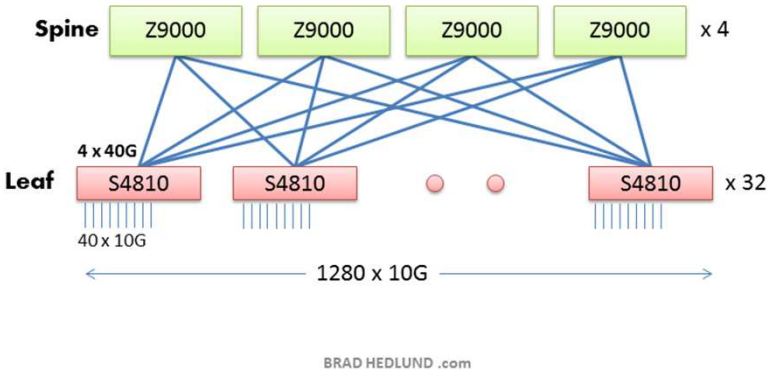
Clos Networks

- Folded Clos: Leaf and Spine
  - Multi-rooted



An Example Clos Network

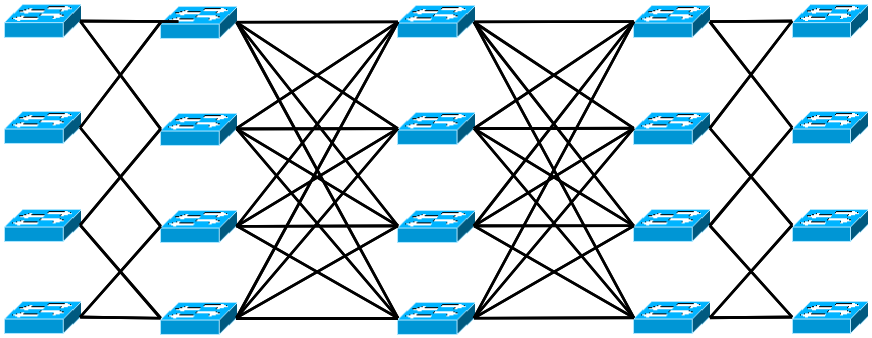
40G Leaf/Spine



Source: https://s3.amazonaws.com/bradhedlund2/2012/40G-10G-leaf-spine/clos-40G.png

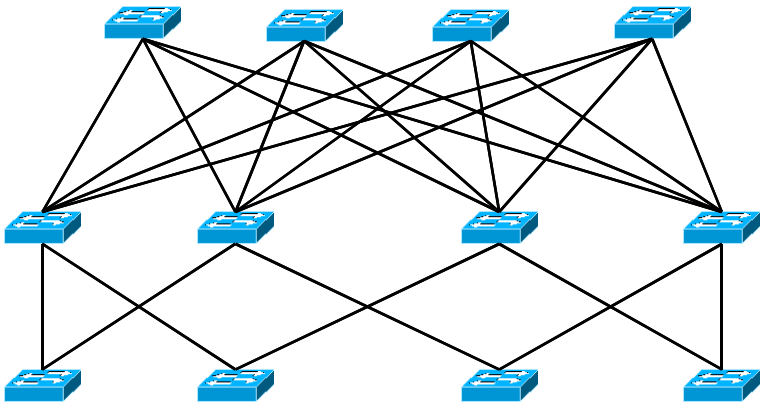
Clos Networks

- 5-stage Clos



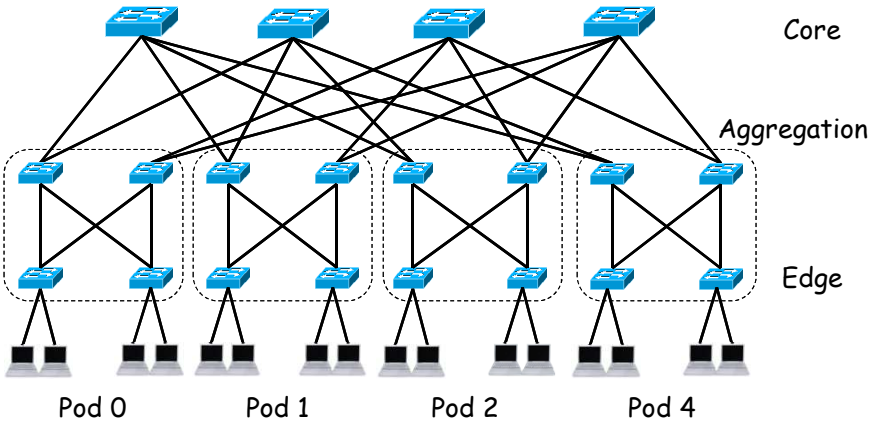
Clos Networks

- Folded Clos: Leaf and Spine



Fat-Tree Topology

("Like trees, they get thicker further from the leaves")



C. E. Leiserson, Fat-trees: Universal networks for hardware-efficient supercomputing, IEEE Trans. Comm. 1985

Fat-Tree Topology

- Fat-Tree: a special type of Clos Network
  - K-ary fat tree: three-layer topology (edge, aggregation, core)
  - Split fat tree into k pods
  - each pod consists of $(k/2)^2$ servers & 2 layers of k/2 k-port switches
  - each edge switch connects to k/2 servers & k/2 aggr. switches
  - each aggr. switch connects to k/2 edge & k/2 core switches
  - $(k/2)^2$ core switches: each connects to k pods
  - Each pod supports non-blocking operation among $(k/2)^2$ hosts
  - With k-port switches, fat tree can support upto $k^3/4$ servers

Oversubscription

- Definition: Ratio of the worst-case aggregate bandwidth required to the total bisection bandwidth of a particular topology
- Bisection bandwidth: sum of bandwidth of smallest set of links that partition the network into two equal halves
- Oversubscription of 1:1: max of 1280 hosts in a single rooted tree with 128-port 10 Gb/s core Ethernet switch
- Oversubscription of 5:1: only 20% of available host bandwidth is available for some communication patterns

Al-Fares et al. 2008 paper

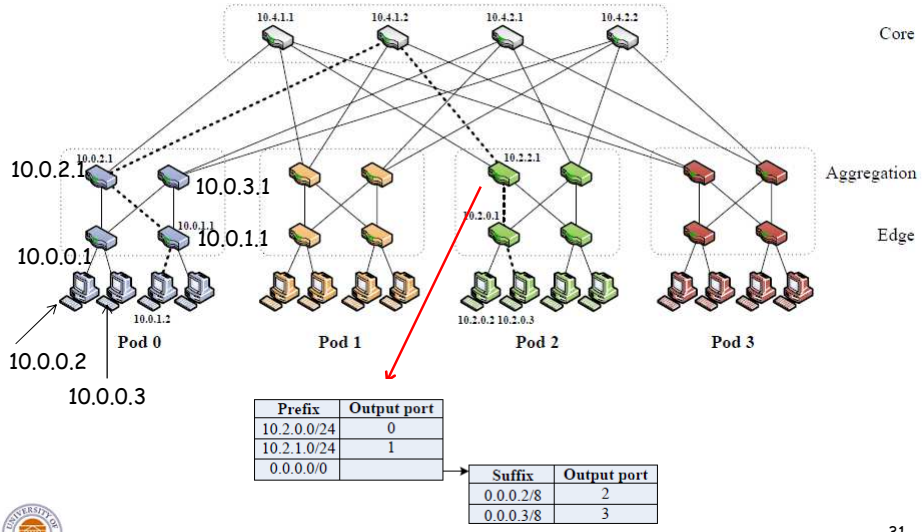
29

Al-Fares et al. 2008 paper

- Asserts: Use commodity switches
- But needs a two-level IP routing table and two-level lookups
- This implies a new implementation of routers is required
- Switch addresses: 10.pod.switch.1
- Host addresses are 10.pod.switch.ID
- pod: 0 to k-1; switch: 0 to k-1 (left-to right, bottom-to-top); ID: 2 to (k/2+1)
- Switch routing tables are created by central controller: given address allocation strategy, algorithmically determined routing tables
- Dynamic routing protocol to handle failures

30

Al-Fares et al. network



Prefix	Output port
10.2.0.0/24	0
10.2.1.0/24	1
0.0.0.0/0	

Suffix	Output port
0.0.0.2/8	2
0.0.0.3/8	3

31

Portland (2009)

- Centralized fabric manager
  - ARP resolution, fault tolerance and multicast
- Hierarchical addressing with MAC addresses: Positional pseudo MAC addresses (PMAC)
- Actual MAC (AMAC) addresses
- Location discovery protocol used to create PMAC based forwarding tables

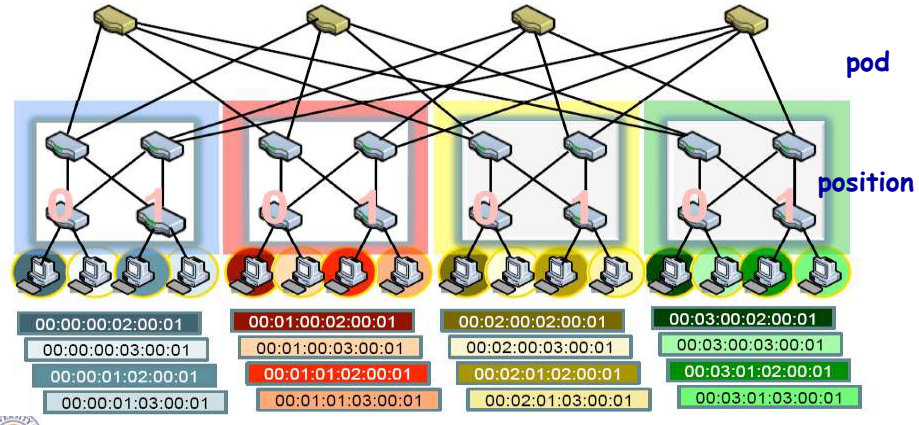
32

Portland
Positional Pseudo MAC Addresses

- Pseudo MAC (PMAC) addresses encodes the location of the host
  - 48-bit: pod.position.port.vmid
  - Pod (16 bit): pod number of the edge switch
  - Position (8 bit): position in the pod
  - Port (8 bit): the port number it connects to
  - Vmid (16 bit): VM id of the host
- Edge switches assign increasing Vmids to each subsequent new MAC address observed on a port

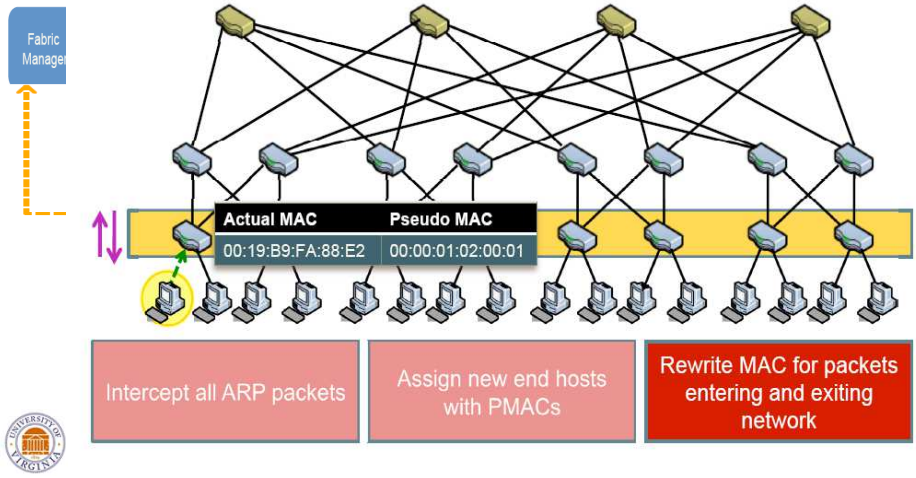
PMAC Addressing Scheme

- PMAC (48 bits): pod.position.port.vmid
  - Pod: 16 bits; position (8 bits); port (8 bits); vmid: 16 bits



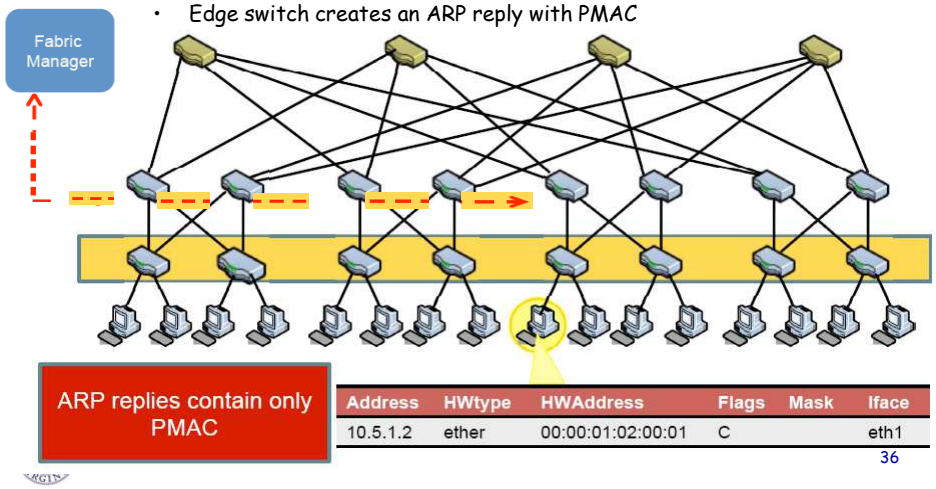
PortLand: PMAC-to-AMAC

- Edge switch listens to end hosts, and discover new source MACs; assigns PMAC addresses; creates its own mapping tables; sends to fabric manager



PortLand: Proxy ARP

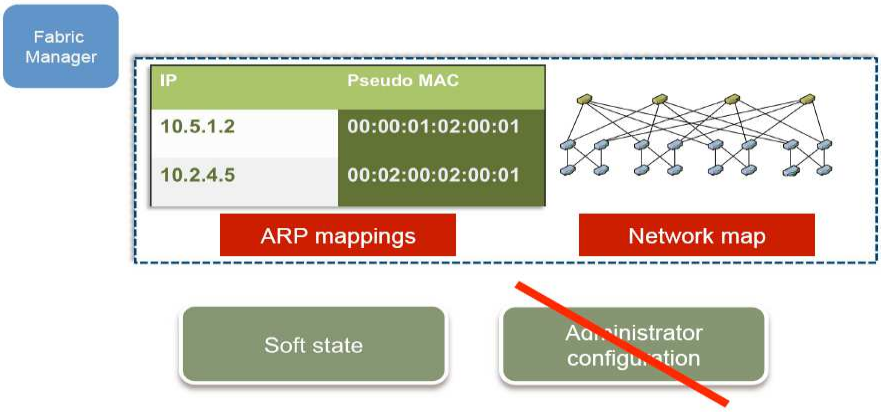
- Edge switch intercepts ARP messages from end hosts and sends request to fabric manager, which replies with PMAC
- Edge switch creates an ARP reply with PMAC



36

PortLand: Fabric Manager

- Fabric manager: logically centralized, multi-homed server
- Maintains topology and <IP,PMAC> mappings in "soft state"



37

Loop Free Forwarding

- When end hosts receive PMAC in ARP response, Ethernet frames created using PMAC addresses in Destination MAC address field
- Forwarding through switches based on PMAC (pod.position.port.vmid)
- Egress edge switch performs PMAC to AMAC rewriting before sending frame on the last hop to the destination host
- Ethernet protocol, frame forwarding and ARP preserved
- Clearly off-the-shelf Ethernet switches cannot be used
- OpenFlow used in prototype implementation

38

Outline

- Introduction
- Challenges in data center networking
- Research papers:
  - Ethernet based
    - ➢ New protocols: DCell, B-Cube
  - Optical, wireless, and energy-efficient architectures
- Standards:
  - IEEE TRILL
  - IEEE 802.1Q
- Summary

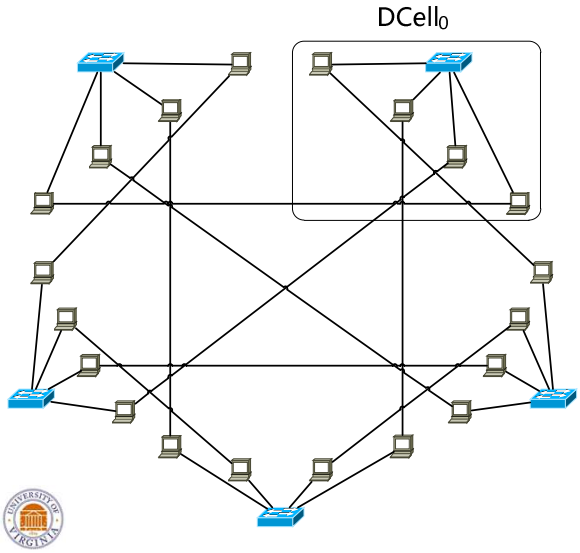
39

Research proposals (new NICs in hosts)

- Guo, C., Lu, G., Li, D.,Wu, H., Zhang, X., Shi, Y., Tian, C.: Zhang, Y., Lu, S., BCube: A high performance, server-centric network architecture for modular data centers. ACM SIGCOMM 2009
- Guo, C., Wu, H., Tan, K., Shi, L., Zhang, Y., Lu, S.: DCell: A scalable and fault-tolerant network structure for data centers. ACM SIGCOMM 2008
- Li, D., Guo, C., Wu, H., Tan, K., Zhang, Y., Lu, S.: Ficonn: Using backup port for server interconnection in data centers. IEEE INFOCOM 2009
- Wu, H., Lu, G., Li, D., Guo, C., Zhang, Y.: MDCube: a high performance network structure for modular data center interconnection. In: Proc. of the 5th intl. conf. on Emerging networking experiments and technologies, ACM 2009

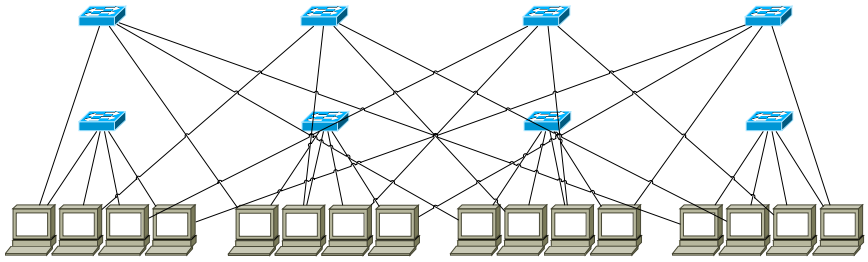
40

DCell



- Recursive design
- Packet forwarding occurs in hosts (see multiple NICs)
- Ethernet switches used just as crossbar switches
- Why?
- Because switches are difficult to program
- Own protocol header
- 32-bit address based
- Hierarchical addressing
- Forwarding is algorithmically determined because of address assignment
- DCell fault-tolerant routing protocol

BCube



- Similar to DCell; switches are crossbars; packet forwarding in servers
- $BCube_k$ is recursively constructed from n $BCube_{k-1}$ and n^k n-port switches; $BCube_0$ is simply n servers connecting to a n-port switch
- Modular Data Center (MDC): shipping container based – easy to move
- BCube packet header sits between Ethernet and IP headers. Fields: source and destination BCube addresses
- One-to-one mapping from IP address to BCube address
- Source routing: complete path stored in header of BCube packets

42

Comparisons

Table 3.1 Summary of Parameters

	Tree-based Architecture			Recursive Architecture	
	Basic Tree	Fat-Tree [2]	Clos Network [11]	DCell [13]	BCube [12]
Degree of Servers	1	1	1	$k+1$	$k+1$
Diameter	$2\log_{n-1}N$	6	6	$2^{k+1}-1$	$\log_n N$
No. of Switches	$\frac{n^2+n+1}{n^3}N$	$\frac{5N}{n}$	$\frac{3}{2}n+\frac{n^2}{4}$	$\frac{N}{n}$	$\frac{N}{n}\log_n N$
No. of Wires	$\frac{n}{n-1}(N-1)$	$N\log_{\frac{n}{2}}\frac{N}{2}$	$N+\frac{4N}{n_{ToR}}$	$\left(\frac{k}{2}+1\right)N$	$N\log_n N$
No. of Servers	$(n-1)^3$	$\frac{n^3}{4}$	$\frac{n^2}{4}\times n_{ToR}$	$\geq (n+\frac{1}{2})^{2^k}-\frac{1}{2}$, $\leq (n+1)^{2^k}-1$	n^{k+1}

[i] Typically k is smaller for DCell because it needs smaller k to connect the same number of servers compared to other recursive-topology architectures.

- N: number of servers; n: no. of ports on the switches; k: no. of levels
- Yang Liu, Jogesh K. Muppala, Malathi Veeraraghavan, Dong Lin, Mounir Hamdi, "Data Center NetworksTopologies, Architectures and Fault-Tolerance Characteristics," SpringerBriefs in Computer Science 2013

Outline

- Introduction
- Challenges in data center networking
- Research papers:
  - Ethernet based
  - New protocols: DCell, B-Cube
  - ➢ Optical, wireless, and energy-efficient architectures
- Standards:
  - IEEE TRILL
  - IEEE 802.1Q
- Summary

Hybrid solutions (w/ optical circuit switches)

- Farrington, N., Porter, G., Radhakrishnan, S., Bazzaz, H., Subramanya, V., Fainman, Y., Papen, G., Vahdat, A.: Helios: a hybrid electrical/optical switch architecture for modular data centers. ACM Sigcomm 2010
- Wang, G., Andersen, D., Kaminsky, M., Papagiannaki, K., Ng, T., Kozuch, M., Ryan, M., c-Through: Part-time optics in data centers, ACM Sigcomm 2010
- Chen, K., Singla, A., Singh, A., Ramachandran, K., Xu, L., Zhang, Y., Wen, X., Chen, Y.: OSA: An optical switching architecture for data center networks with unprecedented flexibility. Usenix NSDI 2012

HELIOS

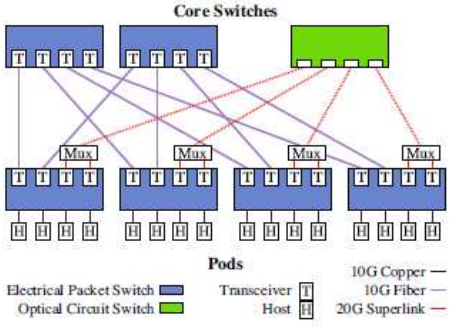


Figure 1: Helios is a 2-level multi-rooted tree of pod switches and core switches. The core consists of both traditional electrical packet switches and MEMS-based optical circuit switches. Superlinks are defined in §3.1.

- pod: shipping container: modular data center
- optical switch: 64-port Glimmerglass MEMS switch
- Monaco 24-port 10 GigE packet switches
- Energy reason: 240 mW/port for optical switch vs. 12.5 W/port in 48-port Arista 7148SW Ethernet switch

HELIOS software

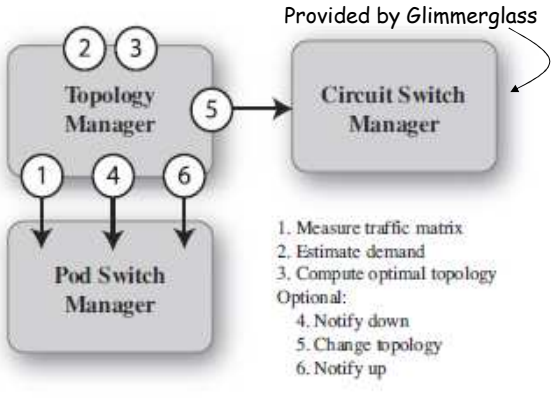


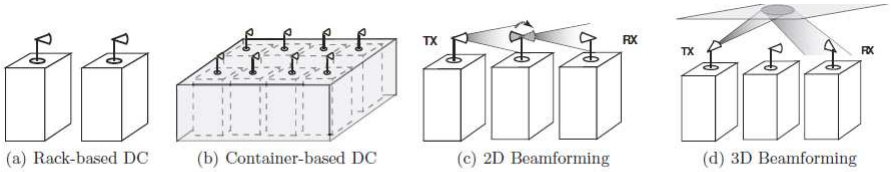
Figure 5: Helios control loop.

- Key problem: Which two pods to connect via optical switch?
- TM: monitors traffic, estimates inter-pod demands, and calculates new circuit configs.
- Mice vs. elephants
- Most of the bandwidth consumed by elephants
- Built and evaluated for some traffic patterns

47

Wireless solutions for data center networks

- W. Zhang, X. Zhou, L. Yang, Z. Zhang, B. Y. Zhao and H. Zheng, 3D Beamforming for Wireless Data Centers, Hotnets2011.
- Key concepts
  - 60 GHz band: Need line-of-sight + signal leakage
  - Hence 3D beamforming
  - Beamforming radios + Ceiling reflectors + EM absorbers



(a) Rack-based DC (b) Container-based DC (c) 2D Beamforming (d) 3D Beamforming

Figure 1: Radio transceivers are placed atop each rack (a) or container (b). Using 2D beamforming (c), transceivers communicate with neighboring racks directly, but forward traffic in multiple hops to non-neighboring racks. Using 3D beamforming (d), the ceiling reflects the signals from each sender to its desired receiver, avoiding multi-hop relays.

48

Benefits and challenges

- Benefits
  - Datacenters of 160 racks and another of 256 racks were used in evaluation
  - Majority of rack pairs could be connected via point-to-point wireless links
  - Use multiple channels to create concurrent links
  - Goal: replace wired cables!
- Challenges
  - Connection management
  - Real-time antenna rotation
  - Physical rack/reflector placement

Energy-efficient datacenters

- D. Abts, M. R. Marty, P. M. Wells, P. Klausler, H. Liu, Energy Proportional Datacenter Networks, ACM ISCA 2010
- Key argument
  - Energy proportional servers consume almost no power when idle and increase consumption as processing load increases
  - As more energy-proportional servers are used in data centers, percent of energy consumption by switches increases
  - Hence need energy proportional switches

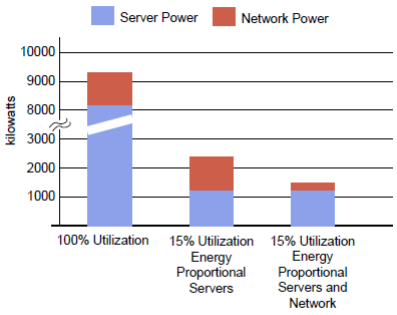


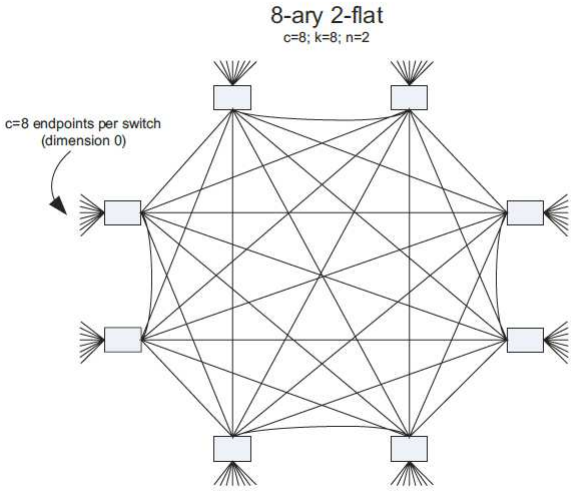
Figure 1: Comparison of server and network power

Key contributions

- Flattened butterfly topology consumes less power than fat tree
- Use high-speed links that support multiple data rates
- Evaluation with traces from a production (Google) datacenter running web search applications.
- Dynamic topologies: change link rates and power up/down links

51

Flattened butterfly



52

General structure for a new datacenter network paper

- Topology?
- Use off-the-shelf switches or new designs?
- Servers with single Ethernet NICs or engage in forwarding?
- Protocol: IP and Ethernet only? Or something new?
- Addressing: hierarchical or flat?
- Address translation: ARP like solutions or other solutions?
- Forwarding: packets sent from node to node
- Routing: computation of forwarding tables (algorithm or protocols)
- OpenFlow switches with SDN controller (centralized)
- Performance enhancements
- Fault tolerance
- Multicast support?
- Implemented?
- How is it evaluated?

Outline

- Introduction
- Challenges in data center networking
- Research papers:
  - Ethernet based
  - New protocols: DCell, B-Cube
  - Optical, wireless, and energy-efficient architectures
- ➢ Standards:
  - IETF TRILL
  - IEEE 802.1Q
- Summary

IETF TRILL (TRansparent Interconnection of Lots of Links)

- Goal
  - Design so that change can be incremental
  - With TRILL, replace any subset of bridges with RBridges (Routing Bridges)
    - still looks to IP like one giant Ethernet
    - the more bridges you replace with RBridges, better bandwidth utilization, more stability

TRILL

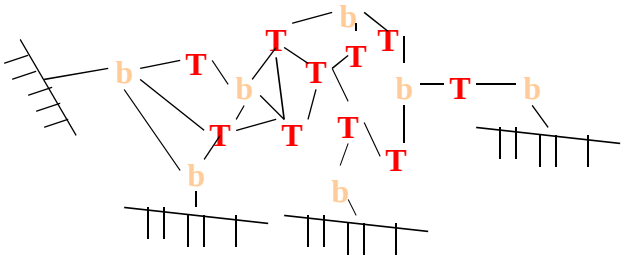
- Basic concept: RBridges (Routing Bridges)
- Use of link-state routing mechanism between RBs
- Frame format
- How addresses are learned?
- Unknown destinations
- [Multicast not covered]

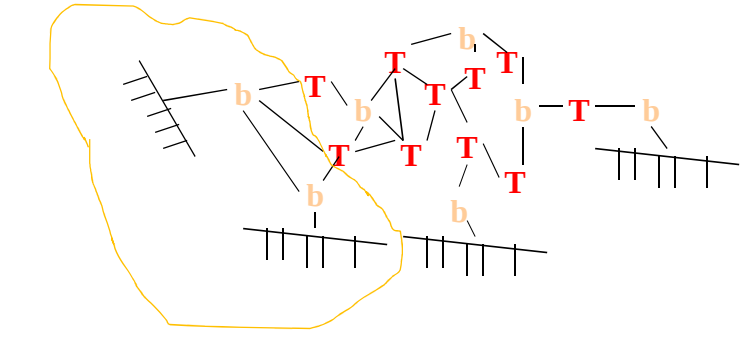
Basic TRILL concept

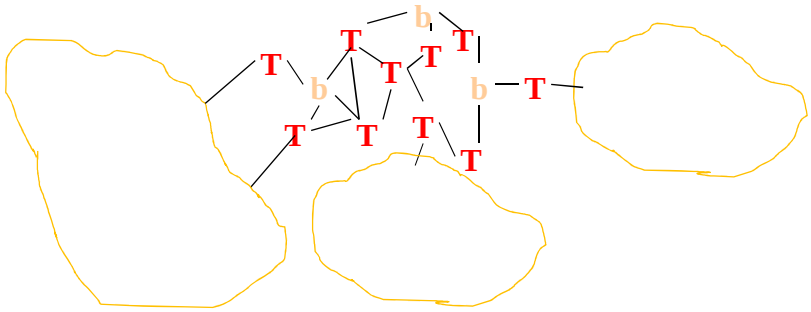
- RBridges find each other (perhaps with bridges in between) with link-state protocol
- Calculate paths to other RBridges
- First RBridge tunnels frames to last RBridge
- Reason for extra header:
  - Forwarding table in RBridges just size of # ofRBridges
  - Layer 3-like header (hop count)
  - Small, easy to look up, addresses

Routing inside campus

- First RB encapsulates frame and sends to last RB
  - So header is "safe" (has hop count – so even if temporary loops are formed, packets will be dropped)
  - Inner RBridges only need to know how to reach destination RBridge
- Still need tree for unknown/multicast
  - But don't need spanning tree protocol – compute tree(s) deterministically from the link state database

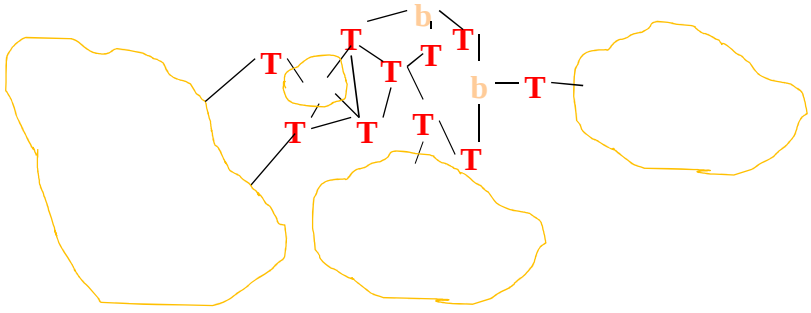

b
T
T
T
T
b
T
b
T
b
T
T
T
T
b
b


b
T
T
T
T
b
T
b
T
b
T
T
T
T
b
b



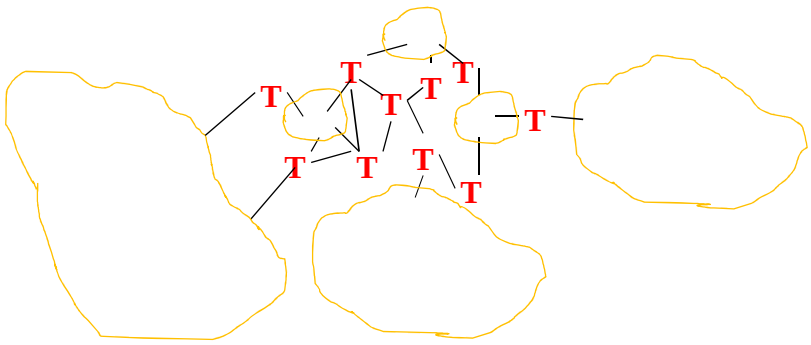
Radia Perlman, Intel Labs, HPSR 2012

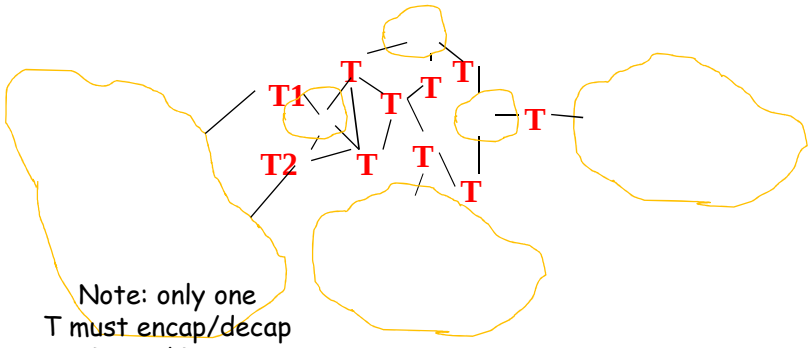
61



Radia Perlman, Intel Labs, HPSR 2012

62





Note: only one T must encap/decap So T1 and T2 must find each other and coordinate (Designated RB)

Frame format

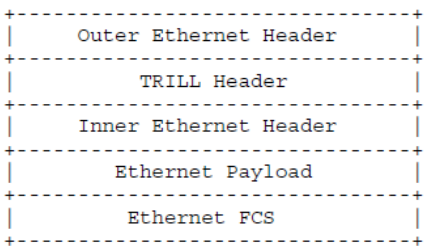


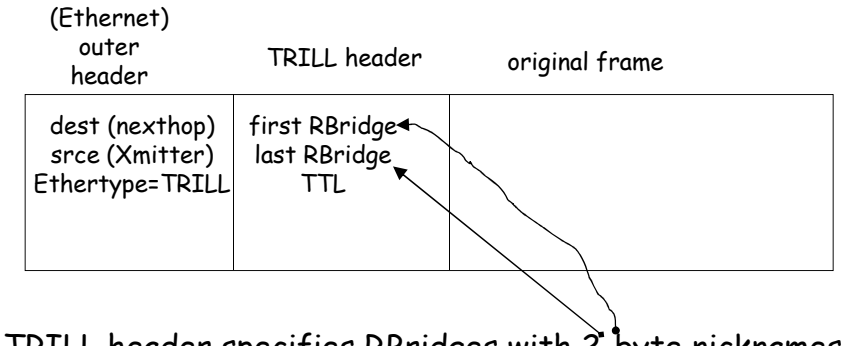
Figure 2: An Ethernet Encapsulated TRILL Frame

- Outer header: MAC addresses of sending and next-hop RBs
- TRILL header: for packet forwarding between ingress-egress RBs
- Inner header: original frame header
- Because TRILL nicknames are not 6-byte MAC, need outer header (compare to PBB)

IETF RFC 6325

65

Encapsulated Frame



TRILL header specifies RBridges with 2-byte nicknames

Radia Perlman, Intel Labs, HPSR 2012

66

2-byte Nicknames

- Saves hdr room, faster fwd'ing
- Dynamically acquired
- Choose unused #, announce in LSP (Link State Protocol: ISIS)
- If collision, IDs and priorities break tie
- Loser chooses another nickname
- Configured nicknames have higher priority

Benefits offered by TRILL header

- loop mitigation through use of a hop count field
- elimination of the need for end-station VLAN and MAC address learning in transit RBridges
- unicast forwarding tables of transit RBridges size depends on the number of RBridges rather than the total number of end nodes
- provision of a separate VLAN tag for forwarding traffic between RBridges, independent of the VLAN of the native frame (inner header VLAN ID different from outer header VLAN ID)

Address learning

- RB1 that is VLAN-x forwarder learns
  - port, VLAN, and MAC addresses of end nodes on links for which it is VLAN-x forwarder from source addresses of frames received
  - Or through configuration
  - Or through Layer-2 explicit registration, e.g., 802.11 Association
- RB1 learns the VLAN and MAC addresses of distant VLAN-x end nodes, and corresponding RB to which they are connected by
  - extracting ingress RB nickname from TRILL header, AND
  - VLAN and source MAC address of the inner frame
- End-Station Address Distribution Information (ESADI) protocol
  - RB that is the appointed VLAN-x forwarder could use this protocol to announce some or all of the attached VLAN-x end nodes to other RBs

Unknown destinations

- If destination address is unknown at an ingress RB, it sends the packets through the spanning tree as an ordinary bridge
- Set the M-bit to 1 (for multicast/broadcast) frames
- For packets sent on links leading to other RBs, it adds a TRILL header and sets the egress RBridge ID to tree ID so that the TRILL frame header is processed by all receiving RBridges on that particular distribution tree

Outline

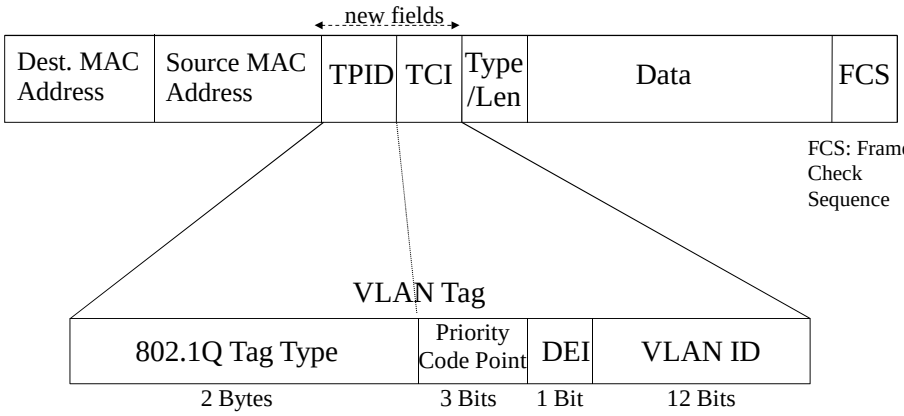
- Introduction
- Challenges in data center networking
- Research papers:
  - Ethernet based
  - New protocols: DCell, B-Cube
  - Optical, wireless, and energy-efficient architectures
- Standards:
  - IEEE TRILL
  - ➢ IEEE 802.1Q: (i) PB/PBB; (ii) SPB; (iii) DCB
- Summary

IEEE bridging protocols

- 802.1D (2004)
  - STP: Spanning Tree Protocol
  - RSTP: Rapid Spanning Tree Protocol (RSTP)
- 802.1Q (2011)
  - VLAN and priority support
  - VLAN classification according to link layer protocol type (802.1v)
  - MSTP: Multiple STP: One STP per non-overlapping group of VLANs (802.1s)
  - Provider bridging (802.1ad)
    - added support for a second level of VLAN tag, called a "service tag", and renamed the original 802.1Q tag a "customer tag". Also known as Q-in-Q because of the stacking of 802.1Q VLAN tags.
  - Provider Backbone Bridges (802.1ah)
    - added support for stacking of MAC addresses by providing a tag to contain the original source and destination MAC addresses. Also known as MAC-in-MAC.

Review from IETF RFC 5556

IEEE 802.1Q Ethernet VLAN



DEI: Drop Eligible Indicator

TPID: Tag Protocol Identifier
TCI: Tag Control Information

Ether type values

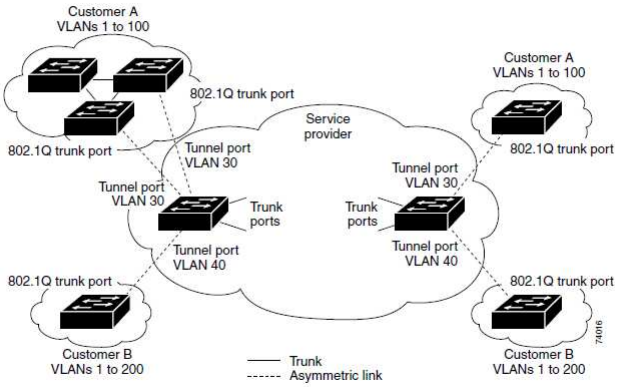
Table 9-1—IEEE 802.1Q EtherType allocations

Tag Type	Name	Value
Customer VLAN Tag	IEEE 802.1Q Tag Protocol EtherType (802.1QTagType)	81-00
Service VLAN Tag or Backbone VLAN Tag	IEEE 802.1Q Service Tag EtherType (802.1QSTagType)	88-a8
Backbone Service Instance Tag	IEEE 802.1Q Backbone Service Instance Tag EtherType (802.1QITagType)	88-e7

- Type field values
  - 0x0800: IP
  - 0x0806: ARP
  - 0x8808: Ethernet flow control (GbE has PAUSE)
  - 0x8870: Jumbo frames (MTU: 9000 Bytes instead of 1500 B)

Provider bridging

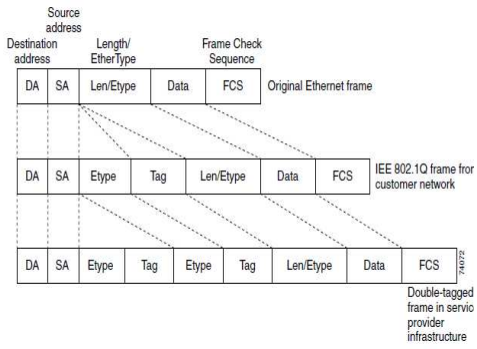
Figure 14-1 802.1Q Tunnel Ports in a Service-Provider Network



- http://www.cisco.com/en/US/docs/switches/metro/me3400e/software/release/12.2_55_se/configuration/guide/swtunnel.pdf

802.1Q and Q-in-Q (provider bridging)

Figure 14-2 Original (Normal), 802.1Q, and Double-Tagged Ethernet Packet Formats



- Frames entering the edge switch tunnel ports with 802.1Q tags are double-tagged when they enter the service-provider network, with the outer tag containing VLAN ID 30 or 40 for customer A and customer B frames, respectively
- Inner tag contains the original customer VLAN number, for example, VLAN 100.
- Both Customers A and B can have VLAN 100 in their networks, the traffic remains segregated within the service-provider network because the outer tag is different.
- Each customer controls its own VLAN numbering space, which is independent of the VLAN numbering space used by other customers and the VLAN numbering space used by the service-provider network.

Provider Bridging (PB) vs. Provider Backbone Bridging (PBB)

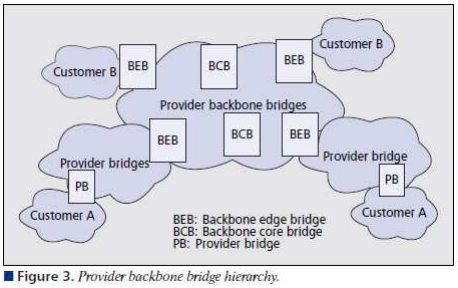


Figure 3. *Provider backbone bridge hierarchy.*

- Fedyk, D.; Allan, D.; , "Ethernet data plane evolution for provider networks [next-generation carrier ethernet transport technologies]," *Communications Magazine, IEEE* , vol.46, no.3, pp.84-89, March 2008

PB and PBB tagging

Type field missing

PB

PBB

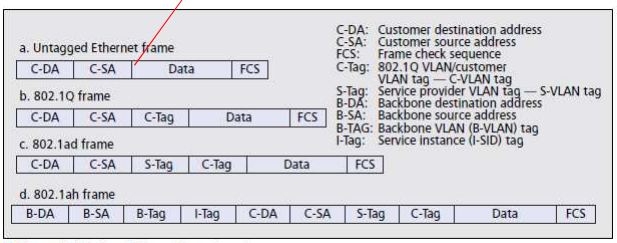


Figure 1. *Various Ethernet frame formats.*

- Salam, S.; Sajassi, A.; , "Provider backbone bridging and MPLS: complementary technologies for next-generation carrier ethernet transport," *Communications Magazine, IEEE* , vol.46, no.3, pp.77-83, March 2008 [Cisco 2008]

Why is PBB required?

- In PB, the service provider network has to learn customer MAC addresses. Hence it is not scalable.
- PBB solves this scalability problem with a new frame format:
  - Customer frame encapsulated in another Ethernet frame with BEB (B-MAC) addresses as source and destination
  - Core switches forward traffic based on backbone MAC (B-MAC) addresses.
  - Confines the requirement to learn customer addresses to the BEB (edge devices) of the PBB network
  - A BEB is required to learn the addresses of only those customers that it supports, and a given BCB is required to learn the addresses of only BEBs (as opposed to having to learn addresses of all of the end customer devices)
  - This greatly enhances the scalability of the solution
- Avaya white paper and Cisco 2008 paper

79

Another problem with PB: service instance scalability: limited to 4096 (12 bit S-VLAN ID)

- PBB frame header: 24-bit I-SID (Backbone Service Instance Identifier)
- Each customer service instance is assigned a unique I-SID value within a service provider's network.
  - Hence, number of service instances increased from 4094 to a theoretical maximum limit of roughly 16 million (2^{24}).
- I-SIDs are visible to BEB (edge) only
- I-SIDs are transparent to the BCB (core)
- PBB frame header also has 12-bit backbone VLAN ID (B-VLAN).
  - Allows provider to partition its network into different broadcast domains
  - Bundle different I-SIDs into distinct B-VLANs
  - Map different B-VLANs into different spanning-tree instances

80

Multi-tenant applications (carrier Ethernet PB, PBB applied to datacenters)

- As large enterprises continue to evolve, many have become very similar to network service providers/carriers. The enterprise IT organization is the "service provider" for its internal customers.
- With the need to support these complex multi-tenant environments comes the added cost and complexity of operating a "carrier-class" network.
- Shortest Path Bridging (SPB) is the technology that will help satisfy all aspects of the multi-tenant customer. The technology evolved from similar protocols used by carriers and service providers. SPB has been enhanced to add "enterprise friendly" features to give it the best of both worlds, carrier robustness / scalability and applicability with enterprise-class features and interoperability.
- http://www.avaya.com/uk/resource/assets/whitepapers/dn4469%20-%20network%20virtual%20using%20spb%20white%20paper.pdf

IEEE 802.1aq Shortest Path Bridging (SPB)

- SPB comes in 2 flavors:
  - SPBV (using 802.1ad aka Q-in-Q)
  - SPBM (using 802.1ah aka MAC-in-MAC encapsulation)
- An SPT Bridge using SPBV mode:
  - supports a C-VLAN or S-VLAN for a single customer
  - uses address learning
- An SPT Bridge using SPBM mode:
  - support B-VLANs in Provider Backbone Bridged Networks
  - does not use source address learning, so unicast B-MAC frames conveying customer data are never flooded throughout the B-VLAN
- Both variants use IS-IS as the link-state routing protocol to compute shortest paths between nodes (RFC 6329)

SPT: Shortest Path Tree

SPB contd.

- Good overview of IEEE 802.1aq in IETF RFC 6329
- IEEE calls it Filtering (of broadcast traffic) databases, while IETF calls it Forwarding (explicit direction of unicast traffic)
- Symmetric (forward and reverse paths) and congruent (with respect to unicast and multicast)
  - shortest path tree (SPT) for a given node is congruent with multicast distribution tree (MDT)
  - preserve packet ordering and share Operations, Administration and Maintenance (OAM) flows with forwarding path
- SPBM filtering database (FDV) is computed and installed for MAC addresses (unicast and multicast)
- SPMV filtering database is computed and installed for VIDs, after which MAC addresses are "learned" for unicast MAC (as in ordinary bridged networks)

Terminology (Multiple Spanning Tree)

- **MST Bridge:** A Bridge capable of supporting the common spanning tree (CST), and one or more MSTIs, and of selectively mapping frames classified in any given VLAN to the CST or a given MSTI.
- **MST Configuration Table:** A configurable table that allocates each and every possible VID to the Common Spanning Tree or a specific Multiple Spanning Tree Instance
- **MST Region:** One or more MST Bridges with the same MST Configuration Identifiers, interconnected by and including LANs for which one of those bridges is the Designated Bridge for the CIST and which have no bridges attached that cannot receive and transmit RST (Rapid Spanning Tree) BPDUs.
- **Multiple Spanning Tree (MST) Configuration Identifier:** A name for, revision level, and a summary of a given allocation of VLANs to Spanning Trees. [New ISIS parameter: 51 B]
- **Multiple Spanning Tree Instance (MSTI):** One of a number of Spanning Trees calculated by MSTP within an MST Region, to provide a simply and fully connected active topology for frames classified as belonging to a VLAN that is mapped to the MSTI by the MST Configuration Table used by the MST Bridges of that MST Region.

IEEE 802.1Q

Routing algorithms

- IETF RFC 6329: Shortest-path default tie-breaking
  - ECT-Algorithm (Equal Cost Tree)
  - Standard ECT Algorithms
- IEEE 802.1Q: Rules for creating MST regions and MSTIs

Industry perspective

- "Industry split on data center network standards" Mar. 22, 2011
  - http://www.networkworld.com/news/2011/032211-trill-ietf-data-center.html
- Vendors
  - Cisco's FabricPath for its Nexus 7000 switch: superset of TRILL
  - BrocadeOne fabric architecture: based on TRILL
  - Juniper:
    - QFabric line: proprietary way of scaling Ethernet in datacenters
  - HP is supporting both TRILL and SPB
  - Huawei is supporting both
  - Avaya and Alcatel-Lucent: supporting SPB given carrier roots

Outline

- Introduction
- Challenges in data center networking
- Research papers:
  - Ethernet based
  - New protocols: DCell, B-Cube
  - Optical, wireless, and energy-efficient architectures
- Standards:
  - IEEE TRILL
  - IEEE 802.1Q: (i) PB/PBB; (ii) SPB; (iii) DCB
- Summary

87

Data Center Bridging (DCB)

- Data Center Bridging is focused primarily on three (3) IEEE specifications:
  - IEEE 802.1Qaz – ETS & DCBX – bandwidth allocation to major traffic classes (Priority Groups); plus DCB management protocol
  - IEEE 802.1Qbb – Priority PAUSE. Selectively PAUSE traffic on link by Priority Group
  - IEEE 802.1Qau – Dynamic Congestion Notification (part of 802.1Q 2011)

Mikkel Hagen, UNH IOL, Data Center Bridging Tutorial
http://www.enterasys.com/company/literature/datacenter-design-guide-wp.pdf

88

IEEE 802.1Qaz

- Enhanced transmission selection
  - Support multiple traffic classes
  - Support priority queueing
  - Support per-traffic class bandwidth allocation (weighted fair queueing)
  - Credit based traffic shaper
- Data Center Bridging eXchange (DCB-X) protocol
  - Discovery of DCB capability in a peer port: for example, it can be used to determine if peer ports support PFC (Priority based Flow Control)
  - DCB feature misconfiguration detection: possible to misconfigure a feature between the peers on a link.
  - Peer configuration of DCB features: if the peer port is willing to accept configuration.

IEEE 802.1Qbb

- Priority based flow control
  - PFC allows link flow control to be performed on a per-priority basis.
  - PFC is used to inhibit transmission of data frames associated with one or more priorities for a specified period of time.
  - PFC can be enabled for some priorities on the link and disabled for others.
- 8 priority levels per port
- In a port of a Bridge or station that supports PFC, a frame of priority n is not available for transmission if that priority is paused on that port.

IEEE 802.1Qau: part of 802.1Q 2011

- Quantized Congestion Notification (QCN) algorithm
  - Congestion Point (CP) Algorithm: a congested bridge samples outgoing frames and generates a feedback message (Congestion Notification Message or CNM) to the source of the sampled frame with information about the extent of congestion at the CP.
  - Reaction Point (RP) Algorithm: a Rate Limiter (RL) associated with a source decreases its sending rate based on feedback received from the CP, and increases its rate *unilaterally* (without further feedback) to recover lost bandwidth and probe for extra available bandwidth.
    - See 802.1Q Section 30 for details
- Congestion Notification Tag
  - An end station may add a Congestion Notification Tag (CN-TAG) to every frame it transmits from a Congestion Controlled Flow (e.g., same src/dst MAC + priority)
  - CN-TAG contains a Flow Identifier (Flow ID) field.
  - The destination_address, Flow ID, and a portion of the frame that triggered the transmission of the CNM are the means by which a station can determine to which RP a CNM applies.

91

Summary

- Challenges in data center networking
  - Neither Ethernet-switched nor IP-routed are ideal
- Research papers:
  - Ethernet based
  - New protocols: DCell, B-Cube
  - Optical, wireless, and energy-efficient architectures
- Standards:
  - IEEE TRILL
  - IEEE 802.1Q: (i) PB/PBB; (ii) SPB; (iii) DCB

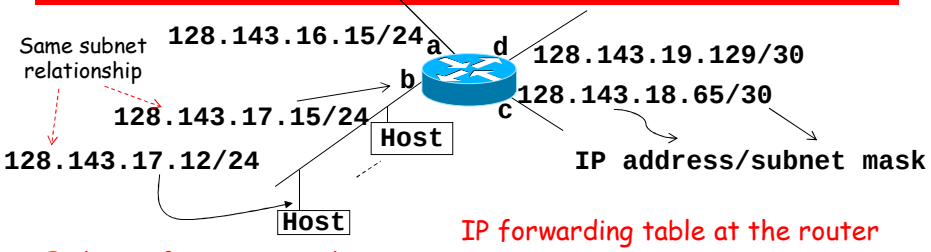
➢ Questions/comments? mvee@virginia.edu

92

Background slides

- IP vs Ethernet
  - Hierarchical vs. Flat addressing
  - Size of forwarding table smaller with hierarchical addressing
  - Hence IP is more scalable making it more suitable for the global Internet
  - Disadv: Need address configuration
- "Network-in-network" for scalability

An IP router with four interfaces



- Each interface is assigned an IP address and a subnet mask by administrator
- The router software will automatically create routing table entries for each subnet as soon as the address/mask are configured for each interface

IP forwarding table at the router

Destination	Subnet mask	Interface (Output port)
128.143.16.0	/24	a
128.143.17.0	/24	b
128.143.18.64	/30	c
128.143.19.128	/30	d

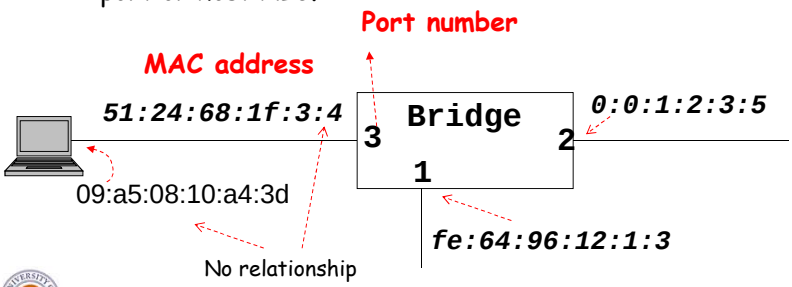
Interpret first two columns of forwarding table as follows

Destination address ranges		Interface
Lower limit	Upper limit	(Output port)
128.143.16.0	128.143.16.255	a
128.143.17.0	128.143.17.255	b
128.143.18.64	128.143.18.67	c
128.143.19.128	128.143.19.131	d

- Routing tables do not store information in this format.
- This format is provided here for ease of learning
- If an incoming datagram destination address is 128.143.16.7, it will be forwarded to output port "a"

MAC addresses

- At a switch (bridge):
  - Each port (interface) has a different MAC address
  - Typically hardwired by switch manufacturer
  - No address assignment by administrator
  - Flat addressing: so any address can be assigned to any switch port or host NIC.

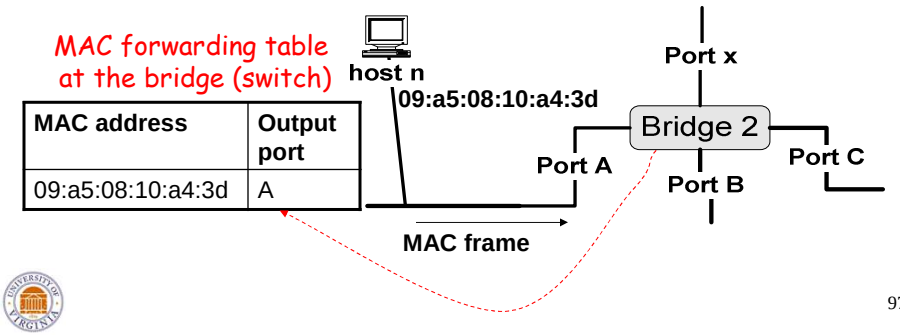


Address Learning: a method for adding entries into the forwarding table

- For each frame received, the bridge stores the source address field in the received frame header into the forwarding table together with the port on which the frame was received.
- A packet destined to a learned address is forwarded to corresponding port.
- Frames with destinations that do not have an entry in the forwarding table are broadcast to all ports (except input port).

MAC forwarding table at the bridge (switch)

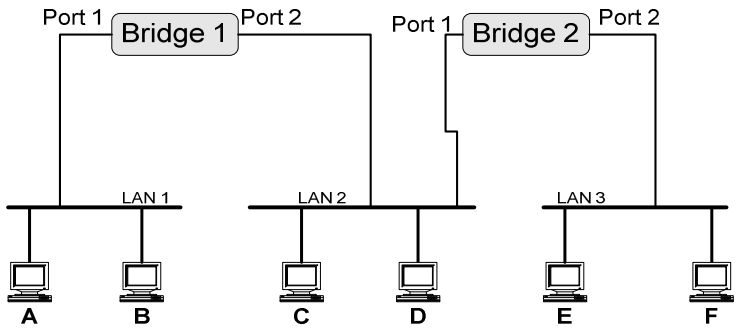
MAC address	Output port
09:a5:08:10:a4:3d	A



Example

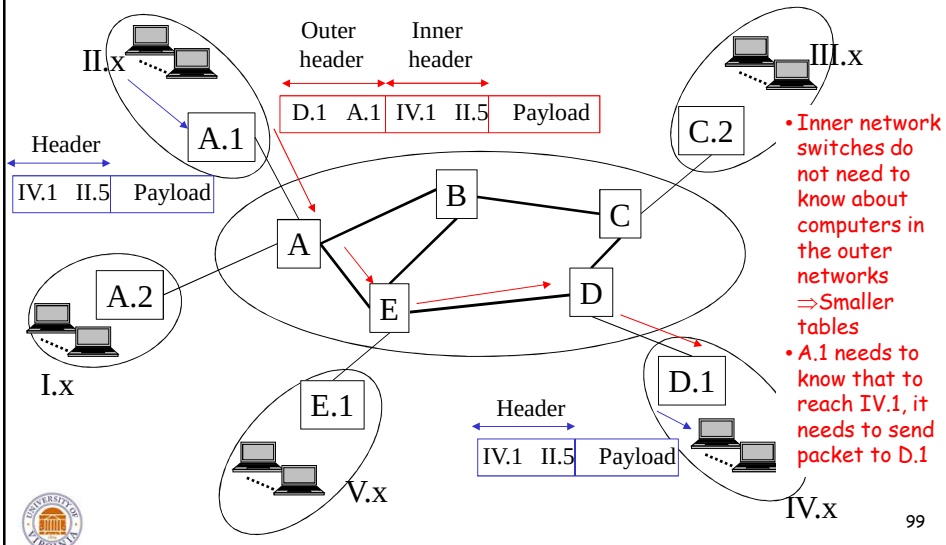
The following frames were sent:
<Src=A, Dest=F>, <Src=C, Dest=A>, <Src=E, Dest=C>

What has each bridge learned?



"Network-in-network"

Outer networks addressing scheme is completely independent of inner network addressing scheme: called map-n-encap



- Inner network switches do not need to know about computers in the outer networks ⇒Smaller tables
- A.1 needs to know that to reach IV.1, it needs to send packet to D.1

99

Applied in many contexts

- IP-in-IP: LISP, shim6 for Internet
- MAC-in-MAC:
  - PBB (Carrrier)
  - SPBM (datacenters)
- VLAN ID-in-VID:
  - PB (Carrier)
  - SPBV (datacenters
- TRILL-in-MAC: TRILL

100